

Date⇒ 09-01-2021

Module⇒ Backend

Lecture By⇒ Akash Handa

Subject ⇒ Blog App

IN PREVIOUS LECTURE (QUICK RECAP) Date-08/02/2021	In Today's Lecture (Overview)
<a href="#">Things Required For The Blogapp</a> <a href="#">Packages Required For The Blog</a>	<a href="#">Express Session</a>

App.js

```
const express = require('express');
const bodyParser = require('body-parser');
const cors = require('cors');
const mongoose = require('mongoose');
const session = require('express-session');
const url = "mongodb://localhost:27017";
const port = process.env.PORT || 6700;
//make object of express so that we can use methods
let app = express();
let dbobj;
app.use(cors());
app.use(bodyParser.urlencoded({extended:true}));
app.use(bodyParser.json());

///use session
app.use(session({
  secret:'mytokensecert'
}));

//static path
app.use(express.static(__dirname+'/public'));
//html
app.set('views','./src/views');
```

```
//view engine
app.set('view engine','ejs');

//we connect with mongo
var MongoClient = new mongodb.MongoClient(url);
MongoClient.connect((err) => {
    if(err) throw err;
    dbobj = MongoClient.db('aryablog')
});

app.get('/health', (req, res) => {
    res.send("Health Ok")
})

app.get('/', (req, res) => {
    let errmessage = req.query.errmessage?req.query.errmessage:'';
    let successmessage = req.query.sucmessage?req.query.sucmessage:'';

    res.render('login', {errmessage:errmessage, successmessage:successmessage})
})

app.get('/register', (req, res) => {
    let errmessage = req.query.errmessage?req.query.errmessage:'';
    res.render('register', {errmessage:errmessage})
})

//Register User
app.post('/register', (req, res) => {
    let user = {
        name:req.body.name,
        email:req.body.email,
        password:req.body.password,
        role:req.body.role?req.body.role:'user',
        isActive:true
    }
    dbobj.collection('users').findOne({email:user.email}, (err, data) => {
        if(data) {
            return res.redirect("/register?errmessage=Email Already Taken Added")
        }else{

```

```

        dbobj.collection('users').insert(user, (err, data) => {
            if(err) throw err;
            //return res.send("Data Added")
            res.redirect("/?sucmessage=Successfully Register")
        })
    }
    })
});

//login
app.post('/login', (req, res) => {
    let user={
        email:req.body.email,
        password:req.body.password
    }
    let count;
    dbobj.collection('users').findOne(user, (err, data) => {
        if(err || !data){
            return res.send('Invalid Login! Please Try Again')
        }else{
            //save data in session
            /*if(req.session.user.email = data.email){

            }*/
            req.session.user = data;
            return res.send(data)
        }
    })
})

//all users
app.get('/allUsers', (req, res) => {
    console.log("session>>>", req.session.user)
    if(!req.session.user){
        return res.send('No Session Found! Please Login Again')
    }
    if(req.session.user.role !== "Admin"){
        return res.send('You are not allowed here')
    }
    dbobj.collection('users').find().toArray((err, data) => {

```

```
        return res.send(data)
      })
    });

//logout
app.get('/logout', (req, res) => {
  req.session.user = null;
  return res.send('Logout Success')
})

app.listen(port, (err) => {
  if(err) throw err;
  console.log(`App is running on port ${port}`)
})
```

## Express Session

```
npm i express-session
```

When implemented, every user of your API or website will be assigned a unique session, and this allows you to store the user state. We'll use the `express-session` module, which is maintained by the Express team. You can install it using `npm install express-session`.

